EtherCAT for motor controllers CMMP-AS

FESTO

Description

EtherCAT
CMMP-AS

Description
570 924
en 0912NH
[749 128]

Original ..... de

Edition ..... en 0912NH

Designation ..... P.BE-CMMX-EC-SW-EN

Order -no. 570 924

© Festo AG & Co. KG, D-73726 Esslingen, 2009
Internet: http://www.festo.com
E-Mail: service_international@festo.com

The copying, distribution and utilisation of this document as well as the communication of its contents to others without express authorization is prohibited. Offenders shall be liable for damages. All rights are reserved, in particular the right to carry out patent, registered design or ornamental design registration.

Adobe and Reader are registered trade marks of Adobe Systems Incorporated in the USA and/or other countries.

Beckhoff and EtherCAT are trademarks of the respective owners in certain countries.

Contents

Intended use VII
Safety instructions VIII
Target group IX
Service IX
Scope of delivery IX
Important user instructions X
Motor controller CMMP-AS documentation XII
Information about the version XIII
Terms and abbreviations XIV

1. Safety instructions for electric drives and controllers 1-1

1.1 General Information 1-3
1.2 Hazards due to improper use 1-6
1.3 Safety instructions 1-7
1.3.1 General safety instructions 1-7
1.3.2 Safety instructions for assembly and maintenance 1-10
1.3.3 Protection against touching electric components 1-13
1.3.4 Protection by low voltage (PELV) against electric shock 1-15
1.3.5 Protection against dangerous movements 1-16
1.3.6 Protection against touching hot components 1-17
1.3.7 Protection when handling and assembling 1-18

2. Overview 2-1

2.1 CANopen 2-3
2.2 EtherCAT 2-5
2.3 EtherCAT protocol 2-10
2.4 EtherCAT CAMC-EC interface 2-11
2.5 XML description file 2-13

3. Assembly 3-1

3.1 Installing the EtherCAT interface in the controller 3-3

4. Installation 4-1

4.1 Pin allocation and cable specifications 4-3
4.1.1 Design of plug connectors X1 and X2 4-3
4.1.2 Allocation of the plug connectors X1 and X2 4-3
4.1.3 Type and layout of cable 4-4
4.1.4 Pin layout for X1 and X2 4-5
4.1.5 Bus termination 4-5

5. CANopen over EtherCAT (CoE) 5-1

5.1 CANopen communication interface 5-3
5.1.1 Configuring the communication interface 5-4
5.1.2 New and revised CANopen communication objects under CoE ... 5-8
5.1.3 Unsupported CANopen communication objects under CoE 5-20
5.2 FHPP profile 5-22
5.3 Communication finite state machine 5-27
5.3.1 Differences between the finite state machines of CANopen and EtherCAT 5-30
5.4 SDO Frame 5-31
5.5 PDO Frame 5-33
5.6 Error Control 5-36
5.7 Emergency Frame 5-37
5.8 Adaptation of the XML device description file 5-38
5.8.1 Fundamental structure of the device description file 5-39
5.8.2 Receive PDO configuration in the RxPDO node 5-41
5.8.3 Transmit PDO configuration in the TxPDO node 5-43
5.8.4 Initialisation commands via the “Mailbox” node 5-44
5.9 synchronisation (Distributed Clocks) 5-46

6. Commissioning 6-1

6.1 Introduction 6-3
6.2 Requirements 6-4
6.3 Important note 6-5
6.4 Step by step 6-6
6.4.1 Configuration using FCT 6-6
6.5 Configuration on a Beckhoff PLC 6-9
6.5.1 Configuration of process data 6-13

7. Service functions and error messages 7-1

7.1 Operating mode and error messages 7-3
7.1.1 Error messages 7-3

A. Technical Appendix A-1

A.1 Technical data A-3

Intended use

This manual is intended to help you work safely with the EtherCAT fieldbus system for the CMMP-AS motor controller. It may only be used in combination with the P.BE-CMMP-AS... product manual.

The CMMP-AS motor controller and the connectable modules and cables may only be used as follows:

– As intended

– Only in an industrial environment

– In perfect technical condition

– In original status without unauthorized modifications (only the conversions or modifications described in the documentation supplied with the product are permitted).

- Observe the safety instructions and use all the components and modules as described in the documentation.
- Observe the standards specified in the relevant chapters, as well as the regulations of the trade associations, the German Technical Control Board (TÜV), the VDE conditions or relevant national regulations.
- Note the maximum limits of all additional components (e.g. sensors, actuators).

Safety instructions

When commissioning and programming positioning systems, the safety regulations in this manual as well as those in the operating instructions for the other components used must be observed.

The user must make sure that nobody is within the sphere of influence of the connected actuators or axis system. Access to the possible danger area must be prevented by suitable measures such as shutting them off and warning signs.

Warning
Electric axes move with high force and at high speed. Collisions can lead to serious injuries and damage to components.

- Make sure that nobody can reach into the sphere of influence of the axes or other connected actuators and that no items are within the positioning range while the system is connected to energy sources.

Warning
Errors in the parametrization can cause injuries and damage to property.

- Enable the controller only if the axis system has been correctly installed and parametrized.

Target group

This manual is intended exclusively for technicians trained in control and automation technology, who have experience in installing, commissioning, programming and diagnosing positioning systems.

Service

Please consult your local Festo Service department or write to the following e-mail address if you have any technical problems:

service_international@festo.com

Scope of delivery

The master configures each slave via an XML file (“EtherCAT Device Description”).
This XML file can be downloaded from the internet at www.festo.com.

Important user instructions

Danger categories

This description contains instructions on the possible dangers which can occur if the product is not used correctly. These instructions are marked (Warning, Caution, etc.), printed on a shaded background and marked additionally with a pictogram. A distinction is made between the following danger warnings:

Warning
... means that failure to observe this instruction may result in serious personal injury or material damage.

Caution
... means that failure to observe this instruction may result in personal injury or material damage.

Note
... means that failure to observe this instruction may result in material damage.

Electrostatically sensitive devices: Inappropriate handling can result in damage to components.

Identification of special information

The following pictograms designate texts that contain special information.

Pictograms

Information:
Recommendations, tips and references to other sources of information

Accessories:
Information on necessary or useful accessories

Environment:
Information on the environmentally-friendly use of the products

Text designations

- Bullet points indicate activities that may be carried out in any sequence.

1. Numerals denote activities which must be carried out in the numerical order specified.

– Hyphens indicate general activities.

Motor controller CMMP-AS documentation

This manual is intended to help you work safely with the **EtherCAT** fieldbus system for the CMMP-AS motor controller.

See the following manuals on the CMMP product family for further information:

Name	Type code	Contents
CANopen manual	P.BE-CMMP-CO-SW...	Description of the implemented CANopen protocol as per CiA 402 and 301.
FHPP manual	P.BE-CMM-FHPP-SW...	Description of the implemented FHPP data profile.
Product manual	P.BE-CMMP-AS...	Description of the technical data and device function as well as information on the installation and operation of the CMMP-AS motor controller.

Information about the version

The information provided in this manual refers to the following hardware and firmware versions of the CMMP-AS motor controller and the following version of the FCT (Festo Configuration Tool) parametrization software:

Firmware	Hardware	Parametrization software	Remarks
3.5.1501.3	from V 1.0	from V 1.2	

The CMMP-AS motor controller has FLASH program memory, which permits the operating software (Firmware) of the CMMP-AS motor controller to be updated after delivery and installation in the machine. The CMMP-AS motor controller operating software is continuously developed and extended by the manufacturer to meet the widest possible range of customer requirements.

Terms and abbreviations

Term / abbreviation	Meaning
ASIC	Application Specific Integrated Circuit
CiA	CAN in Automation
CoE	CANopen over EtherCAT
DS	Draft Standard
ETG	EtherCAT Technology Group
EtherCAT	Ethernet for Controller and Automation Technology
ESC	EtherCAT Slave Controller
FCT	Festo Configuration Tool
FHPP	Festo handling and positioning profile
FPGA	Field Programmable Gate Array
FW	Firmware
IEC	International Electrotechnical Commission
PDI	Process Data Interface

Tab. 0/1: Index of terms and abbreviations

Safety instructions for electric drives and controllers

Chapter 1

Contents

1. Safety instructions for electric drives and controllers 1-1

1.1 General Information 1-3
1.2 Hazards due to improper use 1-6
1.3 Safety instructions 1-7
1.3.1 General safety instructions 1-7
1.3.2 Safety instructions for assembly and maintenance 1-10
1.3.3 Protection against touching electric components 1-13
1.3.4 Protection by low voltage (PELV) against electric shock 1-15
1.3.5 Protection against dangerous movements 1-16
1.3.6 Protection against touching hot components 1-17
1.3.7 Protection when handling and assembling 1-18

1.1 General Information

Festo AG & Co.KG is not liable for damage caused by failure to observe the warning instructions in these operating instructions.

Note
Prior to commissioning, you must read all safety instructions for electric drives and controllers.

If the documentation is not clearly understood in this language, please inform the supplier.

The faultless and reliable operation of the motor controller depends on its correct transport, storage, mounting and installation as well as on careful operation and maintenance.

Note
Only trained and qualified personnel should be allowed to handle the electrical systems.

Trained and qualified personnel

For the purpose of this manual and the warning instructions on the product itself, technicians working with this product must be adequately familiar with the setting up, mounting, commissioning and operation of the product as well as with all warnings and precautionary measures in accordance with the operating instructions in this product manual, and must be sufficiently qualified for this task:

– Training and instructions on or authorization to switch on and switch off devices/systems in accordance with technical safety standards, and to earth and mark them appropriately in accordance with the application requirements.

– Training or instructions in using and maintaining suitable safety equipment in accordance with technical safety standards.

– Training in first aid.

The following instructions must be read before initial commissioning of the system to prevent bodily injuries and/or damage to property.

These safety instructions must be observed at all times.

Do not try to install or commission the motor controller before you have carefully read through all safety instructions for electric drives and controllers in this documentation.
You must read through these safety instructions and all other user instructions before working with the motor controller.

If you do not have any user instructions for the motor controller, please contact your relevant sales representative. Request that the operating instructions are sent immediately to the responsible person, in order that the motor controller can be operated correctly and safely.

These safety instructions must also be provided if the motor controller is sold, lent and/or passed on to third parties.

For safety and guarantee reasons the motor controller may not be opened by the user.

The prerequisite for excellent functioning of the motor controller is professional project planning.

Warning
Incorrect handling of the motor controller and failure to observe the specified warning instructions, as well as improper manipulation of the safety devices can lead to **damage to property, bodily injury, electric shock or in extreme cases, to death**.

1.2 Hazards due to improper use

Warning
High electric voltage and high operating current!
Danger of death or serious bodily injury due to electric shock!

Warning
High electric voltage due to incorrect connection!
Danger of death or bodily injury due to electric shock!

Warning
Surfaces of the device housing may be hot!
Danger of injury! Danger of burning!

Warning
Dangerous movements!
Danger of death, serious bodily injury or damage to property due to unintentional movement of the motors!

1.3 Safety instructions

1.3.1 General safety instructions

Warning
The motor controller complies with protection class IP20, as well as with contamination class 1.

Ensure that the working environment also complies with this protection/contamination class.

Warning
Use only accessories and spare parts which are approved by the manufacturer.

Warning
The motor controllers must be connected to the mains network in accordance with EN standards and VDE regulations, so that they can be disconnected with suitable uncoupling devices (e.g. master switch, fuse, circuit breaker).

The motor controller can be fused with a 300 mA all-current sensitive quick-acting circuit breaker (RCD = Residual Current protective Device).

Warning
Gold contacts or contacts with high contact pressure should be used to connect the control contacts.

Precautionary measures must be taken to prevent interference to switching systems, e.g. Connecting protective switches and relays with RC elements or diodes.

You must observe the safety regulations and directives of the country in which the device is to be used.

Warning
The environmental conditions specified in the product documentation must be observed.

Safety-critical applications are not permitted if unless explicitly approved by the manufacturer.

Instructions for EMC-approved installations can be found in the product manual for the CMMP-AS.
The manufacturer of the system or machine is responsible for ensuring that the limit values required by the national regulations are observed.

Warning
The technical specifications as well as the connection and installation conditions for the motor controller can be found in this product manual. These specifications must be observed at all times.

Warning
The general installation and safety regulations for working on high-current systems (e.g. DIN, VDE, EN, IEC or other national and international regulations) must be observed. Failure to observe these regulations can lead to **bodily injury, death or considerable damage to property**.

The following precautionary measures also apply without claim to completeness:

- VDE 0100 Regulations for setting up high-voltage systems up to 1,000 volts
- EN 60204 Electrical Equipment of Machines
- EN 50178 Equipping high-voltage systems with electronic devices

1.3.2 Safety instructions for assembly and maintenance

When assembling and maintaining the system, the relevant DIN, VDE, EN and IEC regulations, as well as all national and local safety and accident prevention regulations must always be observed. The system manufacturer or the user is responsible for ensuring that these regulations are observed.

Warning
The operation, maintenance and/or commissioning of the motor controller may only be carried out by trained qualified personnel and with electrical appliances suited for this work.

Avoiding accidents, bodily injury and/or material damage:

Warning
The motor holding brake supplied as standard or an external motor-holding brake controlled by the drive controller alone are not suitable for protecting human beings.

- Provide additional support to protect vertical axes from falling or slipping down when the motor is switched off as follows:

– mechanical locking of the vertical axis,
– external braking/safety catch/clamping device or
– sufficient counterbalance of the axis.

Warning
The external or internal brake resistance can cause dangerous intermediate circuit voltage during operation and up to approx. 5 minutes after the motor controller is switched off. If touched, this can lead to fatal or serious injuries.

- Before carrying out maintenance work, make sure that the power supply is switched off and locked and that the intermediate circuit is discharged.
- Disconnect the electrical equipment from the power supply via the main switch and make sure that it cannot be switched on again. Wait until the intermediate circuit is discharged in the following cases:

– Maintenance work and commissioning
– Cleaning
– Extended service interruptions

Warning
Proceed carefully when mounting. During mounting and subsequent operation of the drive, ensure that that no drilling shavings, metal dust or mounting parts (screws, nuts, pieces of wire) fall into the motor controller.

Also make sure that the external power supply of the control section (24 V) is switched off.

Power to the power section must always be switched off before switching off the 24 V power supply to the control section.

Warning
Other work in the vicinity of the machine must only be carried out when the AC or DC supply is switched off and lokked.

Deactivated output stages or deactivated controller enable are not suitable locking conditions. In the event of a fault, this could lead to unintentional movement of the drive.

Warning
Commission the device with a free-running motor, in order to avoid mechanical damage, e.g. due to incorrect direction of rotation.

Warning
Electronic devices are never fail-proof.

- The user is responsible for ensuring that his system is brought into a safe status if the electric device fails.

Warning
The motor controller and, in particular, the brake resistance, external or internal, can become extremely hot, which can cause severe bodily burns on contact.

1.3.3 Protection against touching electric components

This section concerns only devices and drive components with voltages over 50 V. It is dangerous to touch components with voltages of more than 50 V, as this can cause an electric shock. When electric devices are operated, certain components in these devices always carry dangerous voltage.

Warning
Dangerous voltage!

High voltage!

Danger of death or serious bodily injury due to electric shock!

The relevant DIN, VDE, EN and IEC regulations, as well as all national and local safety and accident prevention regulations must always be observed during operation. The system manufacturer or the user is responsible for ensuring that the following regulations are observed:

Warning
Before switching the device on, fit the covers and protective screens so that the device cannot be touched.

For built-in devices, make sure that there is an external housing, such as a control cabinet, to ensure that the electric components cannot be touched.

Regulation VGB4 must be observed.

Warning
Always connect the protective earth conductor of the electrical equipment and the devices firmly to the supply network.

Due to the integrated network filter, the leakage current is greater than 3.5 mA!

Warning
Ensure that the minimum copper cross section is observed for the entire length of the protective earth conductor in accordance with standard EN 60617.

Warning
Before commissioning, also for brief measuring and test purposes, always connect the protective conductor to all electric devices or connect to an earth cable in accordance with the connection diagram.

Otherwise, high voltages may occur on the housing. These could cause an electric shock.

Warning
Do not touch the electrical connection points of the components when the device is switched on.

Warning
- Before touching electric components with voltages over 50 V, disconnect the device from the mains or voltage source.
- Protect the device against being switched on again.

Warning
During installation, note the amount of intermediate circuit voltage, especially with regard to insulation and protective measures.

Make sure that the earthing, the cross section size of the conductor and the corresponding short-circuit protection are correct.

Warning
The device has an intermediate fast discharge circuit in accordance with EN 60204 section 6.2.4. In certain device constellations, particularly where several motor controllers are connected in parallel in the intermediate circuit or with a non-connected braking resistance, the fast discharge may not have any effect. The motor controllers may carry dangerous voltage for up to 5 minutes after being switched off (capacitor residual charge).

1.3.4 Protection by low voltage (PELV) against electric shock

Voltages from 5 to 50 V on the connections and clamps of the motor controller are protective small voltages which can be touched without danger in accordance with the following standards:

international: IEC 60364-4-41

European countries in the EU: EN 50178/1998, Section 5.2.8.1.

Warning
High electric voltage due to incorrect connection!
Danger of bodily injury or death due to electric shock

Devices, electrical components and cables may only be connected to connections and terminals from 0 to 50 V, providing they have protective low voltage (PELV = Protective Extra Low Voltage).

Connect only voltages and circuits with safe separation from dangerous voltages.

Such separation is achieved e.g. via isolating transformers, reliable optocouplers or standalone battery operation.

1.3.5 Protection against dangerous movements

Dangerous movements can be caused by incorrect control of connected motors. The causes can be varied:

– Unsafe or faulty circuitry or cabling

– Faults in operating the components

– Faults in the measured value and signal generators

– Defective or non-EMC compliant components

– Errors in the software in the higher-level control system.

These faults can occur immediately after the device is switched on or after an indeterminate period of operation.

The monitoring functions in the drive components largely exclude the possibility of incorrect operation of the connected drives. With regard to the protection of human beings, especially the danger of bodily injury and/or material damage, one must not rely on this fact alone. Until the fitted monitoring functions become effective, an incorrect drive movement must be expected, the extent of which depends on the type of control and on the operating state.

Warning
Dangerous movements!
Danger of injury or death, serious bodily injury or material damage.

For the above-mentioned reasons, the protection of human beings must be ensured using of monitoring systems or by measures which are of higher order than the system. These measures are incorporated depending on the specific findings of a danger and fault analysis by the system manufacturer. The safety regulations applicable to the system must be observed here as well. Uncontrolled movements of the machine or other incorrect functions can occur as a result of switching off, avoiding or failing to activate safety devices.

1.3.6 Protection against touching hot components

Warning
Surfaces of the device housing may be hot!

Danger of injury! Danger of burning!

Warning
Danger of burning!

- Do not touch the surface of the housing in the vicinity of heat sources.
- After switching devices off, leave them for 10 minutes to cool down before touching them.

If you touch hot parts of the device such as the housing, which contains the heat sink and resistors, you may burn yourself.

1.3.7 Protection when handling and assembling

Handling and assembling certain components in an unsuitable manner can cause injuries under unfavourable circumstances .

Warning
Danger of injury as a result of incorrect handling!

Bodily injury caused by squeezing, shearing, cutting, impact!

The following safety measures apply here:

Warning
- Observe the general regulations on setting up and safety when handling and mounting.
- Use suitable mounting and transport devices.
- Take suitable measures to prevent clamping and crushing.
- Use only suitable tools. If specified, use special tools.
- Use lifting devices and tools in a correct manner.
- If necessary, use suitable protective equipment (e.g. protective glasses, safety shoes, safety gloves).
- Do not stand under hanging loads.
- Wipe up spilt liquids on the floor to avoid slipping.

Overview

Chapter 2

Contents

2. Overview 2-1

2.1 CANopen 2-3
2.2 EtherCAT 2-5
2.3 EtherCAT protocol 2-10
2.4 EtherCAT CAMC-EC interface 2-11
2.5 XML description file 2-13

2.1 CANopen

CANopen is a standard developed by the "CAN in Automation" association. A number of device manufacturers are organised in this association. This standard has largely replaced the current manufacturer-specific CAN protocols. As a result, the end user has a non-proprietary communication interface. The following manuals, among others, can be obtained from this association:

CiA Draft Standard 201-207:
These documents cover the general principles and embedding of CANopen into the OSI layered architecture. The relevant points of this book are presented in this CANopen manual, so procurement of DS201..207 is generally not necessary.

CiA Draft Standard 301:
This book describes the fundamental design of the object directory of a CANopen device and access to it. The statements of DS201..207 are also made concrete. The elements of the object directory needed for the CMMP motor controller families and the related access methods are described in this manual. Procurement of DS301 is recommended but not absolutely necessary.

CiA Draft Standard 402:
deals with the specific implementation of CANopen in drive controllers. Although all implemented objects are also briefly documented and described in this CANopen manual, the user should have this book available.

Source:

CAN in Automation (CiA) International Headquarter
Am Weichselgarten 26
D-91058 Erlangen, Germany
Tel.: 09131-601091
Fax: 09131-601092
www.can-cia.de

The CANopen implementation of the motor controller is based on the following standards:

– [1] - CiA Draft Standard 301, Version 4.02, 13 February 2002

– [2] - CiA Draft Standard Proposal 402, Version 2.0, 26 July 2002

2.2 EtherCAT

The EtherCAT fieldbus system means “Ethernet for Controller and Automation Technology” and was developed by Beckhof Industrie. It is managed by the international EtherCAT Technology Group (ETG) organisation and supports and is designed as an open technology, which is standardised by the International Electrotechnical Commission (IEC).
EtherCAT is a fieldbus system based on Ethernet, which sets new speed standards and can be handled like a fieldbus thanks to flexible topology (line, tree, star) and simple configuration.
The EtherCAT protocol is transported with a specially standardised Ethernet type directly in the Ethernet frame in accordance with IEEE802.3. The slaves can broadcast, multicast and communicate laterally.
For EtherCAT, the data exchange is based on a pure hardware machine. Therefore, special hardware is used on the slave-side, which processes the Ethernet telegram in accordance with the EtherCAT protocol. These hardware protocol interpreters are either provided as ASICs (Application Specific Integrated Circuit), or FPGAs (Field Programmable Gate Array) with corresponding software.

The table below gives an overview of the ASIC and FPGA versions which can be implemented in the hardware.

Function	**ESC10**	**ESC20**	**ASIC**
FMMU	2	4	≥ 8
Sync Manager	4	6	≥ 8
DPRAM	4 kBytes	4 kBytes	4 kBytes
AL Event Mask register		x	x
Process Data Interfaces (PDI)			
8/16 bit µC Interface	x	x	x
32 bit digital	x	x	x

Function	ESC10	ESC20	ASIC
SPI		x	x
Distributed Clocks		x	x

Tab. 2/2: Function overview of the EtherCAT slave controller

The master is generally a pure software solution, which does not require special hardware. That means that in most cases, a standard Ethernet connection is sufficient for the master.

The EtherCAT fieldbus system only defines a new protocol for the transmission layer. It does not define a separate user or device protocol. Instead, EtherCAT can transmit various tried and tested user and device protocols via the EtherCAT protocol (tunnelling).

EtherCAT supports the following user and device protocols:

	supported by CMMP-AS
CANopen over EtherCAT (CoE)	Yes, either with CiA301/402 objects or the Festo FHPP data profile
Servodrive profile to IEC 61491 over EtherCAT (SoE) (corresponds with the SERCOS 2 protocol)	No
File Access over EtherCAT (FoE)	No
Ethernet over EtherCAT (EoE)	No

Tab. 2/3: User and device protocols

The CMMP-AS motor controller with CAMC-EC interface supports the CoE protocol (CANopen over EtherCAT) using FPGA ESC20. This tunnels the CANopen communication objects via the EtherCAT telegram.

In electronic data processing, tunnelling means transmitting data from one network protocol (in this case the CANopen communication objects from the CiA402 and CiA301 protocol), embedded in a different network protocol (in this case, the CoE EtherCAT protocol).

Fig. 2/1 shows the individual protocols of the multiprotocol-capable fieldbus system EtherCAT in a diagram. The CANopen over EtherCAT protocol (CoE protocol) used by the CMMP-AS motor controller is particularly emphasized.

As can be seen in Fig. 2/1, the EtherCAT provides the Mailbox protocol (for noncyclic communication) and the Process Data protocol for exchanging cyclic data on the transmission layer.

These two protocols detect and process the CoE protocol. The CANopen communication objects tunnelled in the EtherCAT protocol are transmitted to the CANopen application level implemented in the CMMP motor controller and internally processed in the motor controller as standard CANopen communication objects.

The Mailbox telegram protocol is used for acyclic data transmission, e.g. for the SDO frame telegram for transmitting standard SDO CANopen communication objects.

The process data telegram protocol is used for cyclic data transmission, e.g. for the PDO frame telegram for transmitting standard PDO telegrams.

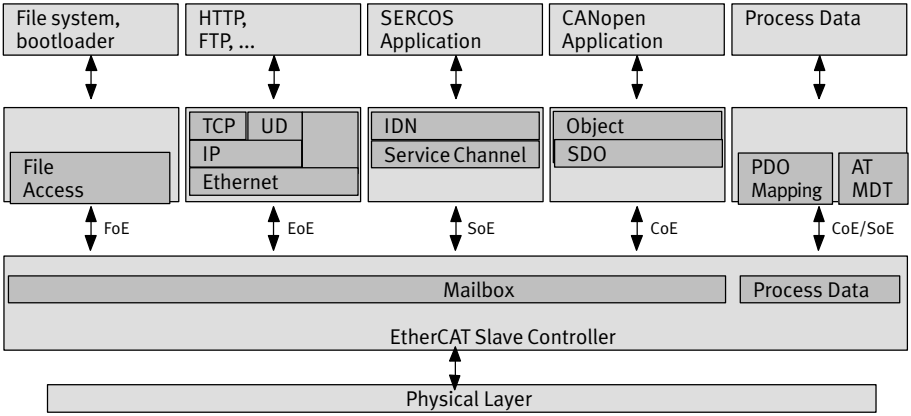

Fig. 2/1: Multiprotocol-capable EtherCAT fieldbus system

The 2.3 “EtherCAT protocol” section describes the EtherCAT protocol in more detail.

The following error detections and diagnostic functions are available to the EtherCAT fieldbus system:

- Broken cable or missing response
- Incorrect reaction in flow (all slaves have not answered)
- Redundant normal operation (second Ethernet port)
  - Wire redundancy
  - Master redundancy with hot standby
  - Device replacement in running network
  - Hot connect of wire segments

For simple interconnection of an EtherCAT compliant device such as the CMMP-AS motor controller, an XML (Extended Markup Language) file is generated for each of these devices. This XML file describes the device to be addressed and its features in detail, and is provided by the manufacturer.
The existing XML file for the CMMP-AS is described in the 2.5 “XML description file” chapter.

2.3 EtherCAT protocol

The EtherCAT protocol is optimised for process data and is transmitted via the Ethernet. A separate EtherType (88A4h) was defined for the EtherCAT protocol. This means that EtherCAT data can be transported directly in standard Ethernet frames. This type of transmission is always selected when all bus stations are in the same Ethernet sub-network. For communication beyond the current sub-network, the UDP/IP protocol can be used to transfer the EtherCAT data packets. For this purpose, the Ethernet header is simply replaced with the UDP/IP header. The EtherCAT user data is unaffected by this(see image Fig. 2/2).
An EtherCAT telegram can consist of multiple sub-telegrams, which can each operate a memory area of the logical image table up to 4 gigabytes in size. The technical data sequence does not depend on the physical sequence of the Ethernet clamps in the network. It can be addressed freely. The slaves can broadcast, multicast and communicate laterally with one another.

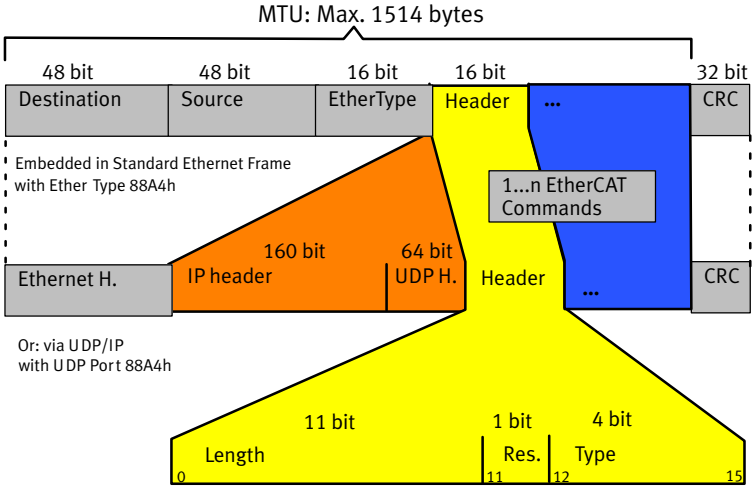

Fig. 2/2: EtherCAT - telegram structure

2.4 EtherCAT CAMC-EC interface

The EtherCAT CAMC-EC interface allows the CMMP motor controller to be connected to the EtherCAT fieldbus system. Communication via the EtherCAT interface (IEEE 802.3u) is implemented using the standard EtherCAT cabling and is possible from CMMP from firmware version 3.5.1501.3.x and the FCT parametrisation software from version 1.2.0.xxx.

Festo supports the CoE protocol (CANopen over EtherCAT) in the CMMP with the Beckhoff FPGA ESC20.
DS402 and FHPP are supported as data profiles.

The EtherCAT interface can only be operated in option slot Ext2. Other interface modules cannot then be operated in option slot Ext1 unless the CAMC-D-8E8A module is used.

EtherCAT CAMC-EC interface characteristics

The EtherCAT interface has the following features:

- Can be mechanically fully integrated in the CMMP-AS series motor controllers
- EtherCAT conforms to IEEE-802.3u (100Base-TX) at 100Mbps (full duplex)
- Star and line topology
- Plug connectors: RJ45
- Electrically isolated EtherCAT interface
- Communication cycle: 1 ms (for DS402 profile); 2 ms (for FHPP profile)
- Max. 127 slaves
- EtherCAT slave implementation based on the Beckhoff FPGA ESC20

- Support of the ”Distributed Clocks” feature for time synchronous setpoint value transfer
- LED displays for ready status and link detect

2.5 XML description file

In order to connect EtherCAT slave devices simply to an EtherCAT master, there must be a description file for every EtherCAT slave device. This description file is comparable with the EDS files for the CANopen fieldbus system or the GSD files for Profibus. In contrast to the latter, the EtherCAT description file is in the XML format, as is often used for internet and web applications, and contains information on the following features of the EtherCAT slave devices:

– Information on the device manufacturer

– Name, type and version number of the device

– Type and version number of the protocol to be used for this device (e.g. CANopen over Ethernet, ...)

– Parametrisation of the device and configuration of the process data

EXAMPLE of an XML description file for the CMMP:

```
‹?xml version=”1.0” encoding=”UTF-8”?›
‹EtherCATInfo Version=”0.2”›
   ‹Vendor›
      ‹Id›#x1D‹/Id›
      ‹Name›Festo AG‹/Name›
      ‹ImageData16x14›424DE60000000000000076000000
      28000000100000000E00000001000400000000007000
      0000C40A0000C40A00000000000000000000BA79230
      0CE962900CC944300DCAD4900D2A36600E2B96900E3
      C07E00EED09800F3D7AE00F3E0BF00F7EBCA00F7EED
      800F7EF00FFF5ED00FFFFF700FFFFFF00ECFDCCCCBDE
      BECBDB6F9445448D4B647A4F7223126C09225A3E67
      B8585B085A3A3A66AA875C276E3A146448625C276E3
      B006206005C076E3A3A58B84ABC076D1A2856772767
      16371A033136233315333B013113307214507D99A99
      9A8C99AB8CFEEEEEEEEFEEEEEFFFFFFFFFFFFFFFF
      ‹/ImageData16x14›
   ‹/Vendor›
‹Descriptions›
```

```
    ‹Groups›
        ‹Group SortOrder=”1”›
        ‹Type›Festo E-Drives‹/Type›
        ‹Name LcId=”1033”›Festo E-Drives‹/Name›
    ‹/Groups›
‹Devices›
    ‹Device Physics=”YY”›
    ‹Type ProductCode=”#x264860d3” RevisionNo=
    ”#x20000”›CMMP-AS
    ‹/Type›
    ‹Name LcID=”1033”›CMMP-AS‹/Name›
    ‹Name LcID=”1031”›CMMP-AS‹/Name›
```

If the EtherCAT protocol is used, the communication partner for the used device can be parametrised completely via the XML description file. For example, it includes parametrisation of the process data objects for cyclic data exchange when CoE is used.

This data is read during the initialisation phase of the EtherCAT master (generally a PLC or CNC) and used for the initialisation and cyclic communication with the slave device.

Assembly

Chapter 3

Contents

3. Assembly 3-1

3.1 Installing the EtherCAT interface in the controller 3-3

3.1 Installing the EtherCAT interface in the controller

Warning
Non-compliance with the instructions in Chapter 1 ”Safety instructions for electric drives and controllers” **can result in material damage, injury, electric shock, or in extreme cases, even to fatalities**.

Warning
The drive controller must be disconnected from all current carrying cables before connecting the module.

Use a suitable Phillips screwdriver to remove the front cover over option slot Ext2 of the CMMP motor controller. The EtherCAT technology module is now placed in the option slot (Ext2) so that the printed circuit board slides into the guides on the sides of the optional slot. Push the technology module in as far as possible. Then screw the technology module to the motor controller housing using the Phillips screw. Ensure that the front panel has conducting contact to the CMMP housing (protective earth/PE).

The following elements can be found on the front panel of the EtherCAT technology module:

- **LED 1** (two-colour LED) for:
  - EtherCAT communication (yellow)
  - “Connection active on Port 1” (red)
  - Run (green)
- **LED 2** (red) for displaying “Connection active on Port 2”
- Two RJ45 sockets.

The following figure shows the position and numbering of the sockets:

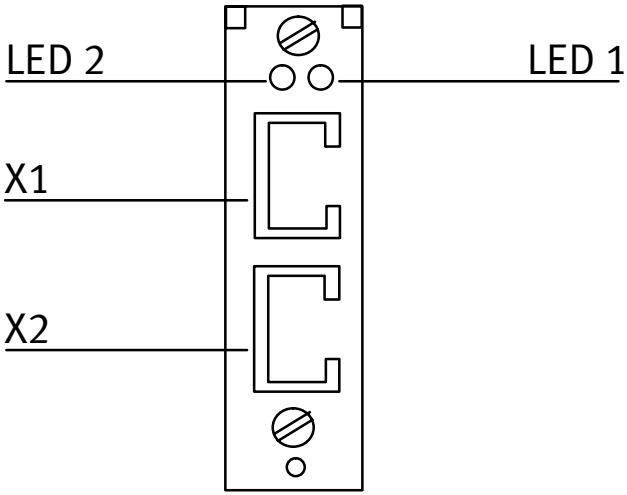

Fig. 3/3: Position of the elements on the front panel

Installation

Chapter 4

Contents

4. Installation 4-1

4.1 Pin allocation and cable specifications 4-3
4.1.1 Design of plug connectors X1 and X2 4-3
4.1.2 Allocation of the plug connectors X1 and X2 4-3
4.1.3 Type and layout of cable 4-4
4.1.4 Pin layout for X1 and X2 4-5
4.1.5 Bus termination 4-5

4.1 Pin allocation and cable specifications

4.1.1 Design of plug connectors X1 and X2

	RJ45 sockets
X1 (RJ45 socket at top)	Uplink to master or a previous station of a series connection (e.g. multiple motor controllers)
X1 (RJ45 socket at bottom)	Uplink to master, end of a series connection or connection of additional downstream stations

4.1.2 Allocation of the plug connectors X1 and X2

PIN	**Specification**	
1	Receiver signal- (RX-)	Wire pair 3
2	Receiver signal+ (RX+)	Wire pair 3
3	Transmission signal- (TX-)	Wire pair 2
4		Wire pair 1
5		Wire pair 1
6	Transmission signal+ (TX+)	Wire pair 2
7		Wire pair 4
8		Wire pair 4

Tab. 4/4: Allocation of the plug connectors X1 and X2

4.1.3 Type and layout of cable

Shielded twisted-pair STP, Cat.5 cables must be used for cabling. Star and line topologies are supported. The network structure must conform to the 5-4-3 rule. No more than 10 hubs can be cabled in series. The EtherCAT technology module contains a hub. The total cable length is restricted to 100 m.

The cable names stated refer to cables made by LAPP and Lütze. They have proven themselves in practice and are successfully in use in many applications. However, comparable cables by other manufacturers can also be used.

Cable length	**Order number**
LAPP EtherCAT cable	
0.5 m	90PCLC50000
1 m	90PCLC500010
2 m	90PCLC500020G
5 m	90PCLC500050G
Lütze EtherCAT cable	
0.5 m	192000
1 m	19201
5 m	19204

Tab. 4/5: EtherCAT cable

Caution
Errors due to unsuitable bus cable

As very high baud rates can occur, we recommend that you use only the standardised cables and plug connectors. In some cases, they have additional diagnostics options and allow the fieldbus interface to be analysed rapidly in the event of errors.

When setting up the EtherCAT network, you must follow the advice in the relevant literature or the following information and instructions in order to maintain a stable, trouble free system. The system is not cabled properly, EtherCAT bus malfunctions can occur during operation. These can cause the CMMP motor controller to shut off with an error for safety reasons.

4.1.4 Pin layout for X1 and X2

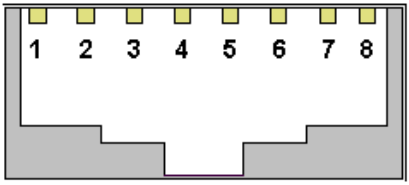

Fig. 4/4: Pin layout for X1 and X2

4.1.5 Bus termination

No external bus terminations are required. The EtherCAT technology module monitors its two ports and terminates the bus independently (loop-back function).

CANopen over EtherCAT (CoE)

Chapter 5

Contents

5. CANopen over EtherCAT (CoE) 5-1

5.1 CANopen communication interface 5-3
5.1.1 Configuring the communication interface 5-4
5.1.2 New and revised CANopen communication objects under CoE ... 5-8
5.1.3 Unsupported CANopen communication objects under CoE 5-20
5.2 FHPP profile 5-22
5.3 Communication finite state machine 5-27
5.3.1 Differences between the finite state machines of CANopen and EtherCAT 5-30
5.4 SDO Frame 5-31
5.5 PDO Frame 5-33
5.6 Error Control 5-36
5.7 Emergency Frame 5-37
5.8 Adaptation of the XML device description file 5-38
5.8.1 Fundamental structure of the device description file 5-39
5.8.2 Receive PDO configuration in the RxPDO node 5-41
5.8.3 Transmit PDO configuration in the TxPDO node 5-43
5.8.4 Initialisation commands via the “Mailbox” node 5-44
5.9 synchronisation (Distributed Clocks) 5-46

5.1 CANopen communication interface

As already described in Chapter 1 user protocols are tunnelled via EtherCAT. For the CANopen over EtherCAT protocol (CoE) supported by the CMMP-AS, most objects for the communication layer are supported by EtherCAT in accordance with CANopen standard DS301 of the CiA. This primarily involves objects for setting up communication between masters and slaves.

For the CANopen Motion Profile in accordance with CiA 402, most objects which can also be operated via the standard CANopen fieldbus, are supported. In general, the following services and object groups are supported by the EtherCAT CoE implementation in the CMMP-AS motor controller:

SDO	Service Data Object: Used for normal parametrisation of the motor controller.
PDO	Process Data Object: Fast exchange of process data (e.g. actual speed) possible.
EMCY	Emergency Message: Transmission of error messages.

The individual objects which can be addressed via the CoE protocol in the CMMP-AS motor controller are internally forwarded to the existing CANopen implementation and processed there. For this reason, the individual CANopen objects are not listed again in this manual. Please refer to the CANopen manual.

The individual CANopen communication objects are described in the ”CMMP-AS Motor Controller” CANopen manual.

However, some new CANopen objects are added under the CoE implementation under EtherCAT, which are required for special connection via CoE. This is the result of the revised communication interface between the EtherCAT protocol and the CANopen protocol. A so-called Sync Manager is used to control the transmission of PDOs and SDOs via the two EtherCAT transfer types (mailbox and process data protocol).

This Sync Manager and the necessary configuration steps for operating the CMMP-AS under EtherCAT-CoE are described in the 5.1.1 “Configuring the communication interface” Chapter. The additional objects are described in the 5.1.2 “New and revised CANopen communication objects under CoE” chapter.

Also, some CANopen objects of the CMMP-AS, which are available under a normal CANopen connection, are not supported via a CoE connection over EtherCAT. The 5.1.3 “Unsupported CANopen communication objects under CoE” contains a list of CANopen objects not supported under CoE.

5.1.1 Configuring the communication interface

As already described in Chapter 1, the EtherCAT protocol uses two different transfer types to transmit the device and user protocols, e.g. the CANopen over EtherCAT protocol(CoE) used by CMMP-AS. The two transfer types are the mailbox telegram protocol for noncyclic data and the process data telegram protocol for transmitting cyclic data.

These two transfer types are used for the different CANopen transfer types for the CoE protocol. They are used as follows:

– Mailbox telegram protocol:
  This transfer type is used to transmit the Service Data Objects (SDOs) defined under CANopen. They are transmitted to EtherCAT in SDO frames. They are described in more detail in the 5.4 “SDO frame” chapter.

– Process data telegram protocol:
  This transfer type is used to transmit the Process Data Objects (PDOs) defined under CANopen, which are used to exchange cyclic data. They are transmitted to EtherCAT in PDO frames. They are described in more detail in the 5.5 “PDO frame” chapter.

In general, these two transfer types allow all PDOs and SDOs to be used exactly as they are defined for the CANopen protocol for CMMP-AS. Refer to the CANopen manual for the CMMP-AS motor controller.

However, parametrisation of PDOs and SDOs for sending objects via EtherCAT is different to the settings which must be made under CANopen. In order to link the CANopen objects to be exchanged via PDO or SDO transfers between masters and slaves into the EtherCAT protocol, a so-called Sync Manager is implemented under EtherCAT.

This Sync Manager is used to link the data of the PDOs and SDOs to be sent to the EtherCAT telegrams. To accomplish this, the Sync Manager provides multiple Sync channels which can each implement a CANopen data channel (Receive SDO, Transmit SDO, Receive PDO or Transmit PDO) on the EtherCAT telegram.

The figure shows how the Sync Manager is linked to the system:

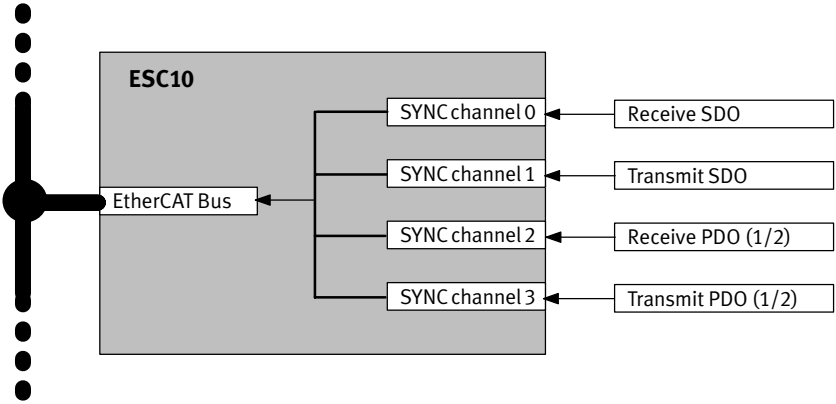

Fig. 5/5: Sample mapping of the SDOs and PDOs to the Sync channels

All objects are sent via so-called Sync channels. The data from these channels is automatically linked to the EtherCAT data flow and transmitted. The EtherCAT implementation in the CMMP-AS motor controller supports four such Sync channels.

For this reason, additional mapping of the SDOs and PDOs to the Sync channels is required compared with CANopen. This is implemented via the so-called Sync Manager objects (objects 0x1C00 and 0x1C10 to 0x1C13 / see Chapter 5.1.2). These objects are described in more detail below.

The CMMP-AS has four individual Sync channels. These Sync channels are permanently allocated to the individual transfer types and cannot be changed by the user. The allocation is as follows:

– Sync channel 0: Mailbox telegram protocol for incoming SDOs (Master => Slave)

– Sync channel 1: Mailbox telegram protocol for outgoing SDOs (Master <= Slave)

– Sync channel 2: Process data telegram protocol for incoming PDOs (Master => Slave) Object 0×1C12 must be observed.

– Sync channel 3: Process data telegram protocol for outgoing PDOs (Master <= Slave) Object 0×1C13 must be observed.

The parametrisation of the individual PDOs is set via objects 0x1600 to 0x1603 (Receive PDOs) and 0x1A00 to 0x1A03 (Transmit PDOs). The PDOs are parametrisation as described in the CANopen manual.

The Sync channels can only ever be set and the PDOs only configured in the ”Pre-Operational” status.

EtherCAT does not allow the slaves to parameterise themselves. The device description files are available for this purpose. They prescribe the entire parametrisation, including PDO parametrisation, which is used by the master during initialisation.

All changes to the parametrisation should therefore not be made by hand, but in the device description files. For this purpose, important sections of the device description files are described in more detail in Chapter 5.8.

The Sync channels described here are NOT the same as the Sync telegrams familiar from CANopen. CANopen Sync telegrams can still be transmitted as SDOs via the SDO interface implemented under CoE, but do not directly influence the Sync channels described above.

5.1.2 New and revised CANopen communication objects under CoE

The following table contains an overview of the indices and subindices used for CANopen compatible communication objects, which are inserted in the range from 0×1000h to 0×1FFFh for the EtherCAT fieldbus system. They primarily replace the communication parameters in accordance with CiA 301.

Value	Meaning	Permitted with
0x1000	Device type	Device control identifier
0x1018	Identity object	Vendor ID, product code, revision. serial no.
0x1100	EtherCAT fixed station address	Fixed address assigned to the slave during initialisation by the master
0x1600	1. RxPDO Mapping	Identifier of the 1st receive
0x1601	2. RxPDO Mapping	Identifier of the 2nd receive
0x1602	3. RxPDO Mapping	Identifier of the 3rd receive
0x1603	4. RxPDO Mapping	Identifier of the 4th receive
0x1A00	1. TxPDO Mapping	Identifier of the 1st transmit
0x1A01	2. TxPDO Mapping	Identifier of the 2nd transmit
0x1A02	3. TxPDO Mapping	Identifier of the 3rd transmit
0x1A003	4. TxPDO Mapping	Identifier of the 4th transmit
0x1C00	Sync Manager Communication Type	Object for configuring the individual Sync channels (SDO or PDO Transfer)
0x1C10	Sync Manager PDO Mapping for Sync Channel 0	Assignment of the Sync channel 0 to a PDO/SDO (Channel 0 is always reserved for the Mailbox Receive SDO Transfer)
0x1C11	Sync Manager PDO Mapping for Sync Channel 1	Assignment of the Sync channel 1 to a PDO/SDO (Channel 1 is always reserved for Mailbox Send SDO Transfer)
0x1C12	Sync Manager PDO Mapping for Sync Channel 2	Assignment of the Sync channel 2 to a PDO (Channel 2 is reserved for Receive PDOs)
0x1C13	Sync Manager PDO Mapping for Sync Channel 3	Assignment of the Sync channel 3 to a PDO (Channel 3 is reserved for Transmit PDOs)

Tab. 5/6: New and revised communication objects

The subsequent chapters describe the objects 0×1C00 and 0×1C10 to 0×1C13 in more detail, as they are only defined and implemented under the EtherCAT CoE protocol and therefore are not documented in the CANopen manual for the CMMP-AS motor controller.

The CMMP-AS motor controller with the EtherCAT interface supports four Receive PDOs (RxPDO) and four Transmit PDOs (TxPDO).

Objects 0x1008, 0x1009 and 0x100A are not supported by CMMP-AS, as plain text strings cannot be read from the motor controller.

Object 1100h - EtherCAT fixed station address

This object allocates a unique address to the slave during the initialisation phase. The object has the following meaning:

Index	**1011h**
Name	EtherCAT fixed station address
Object code	V
Data Type	uint16
Access	ro
PDO Mapping	no.
Value Range	0...FFFFh
Default Value	0

Object 1C00h - Sync Manager Communication Type

This object allows the transfer type for the various channels of the EtherCAT Sync Manager to be read. As the CMMP-AS only supports the first four Sync channels under the EtherCAT CoE protocol, the following objects are “read only”.
That means that the configuration of the Sync Manager for CMMP-AS is fixed. The objects have the following meaning:

Index	**1C00h**
Name	Sync Manager Communication Type
Object Code	Array
Data Type	uint8

Sub-Index	**00h**
Description	Number of Used Sync Manager Channels
Access	ro
PDO Mapping	no.
Value Range	4
Default Value	4

Sub-Index	**01h**
Description	Communication Type Sync Channel 0
Access	ro
PDO Mapping	no.
Value Range	2: Mailbox Transmit (Master => Slave)
Default Value	2: Mailbox Transmit (Master => Slave)

Sub-Index	**02h**
Description	Communication Type Sync Channel 1
Access	ro
PDO Mapping	no.
Value Range	2: Mailbox Transmit (Master <= Slave)
Default Value	2: Mailbox Transmit (Master <= Slave)

Sub-Index	**03h**
Description	Communication Type Sync Channel 2
Access	ro
PDO Mapping	no.
Value Range	0: Unused 3: Process Data Output (RxPDO / Master => Slave)
Default Value	3

Sub-Index	**04h**
Description	Communication Type Sync Channel 3
Access	ro
PDO Mapping	no.
Value Range	0: Unused 4: Process Data Input (TxPDO / Master <= Slave)
Default Value	4

Object 1C10h - Sync Manager Channel 0 (Mailbox Receive)

This object allows a PDO to be configured for Sync channel 0. As Sync channel 0 is always allocated to the mailbox telegram protocol, the user cannot change this object. The object therefore always has the following values:

Index	**1C10h**
Name	Sync Manager Channel 0 (Mailbox Receive)
Object Code	Array
Data Type	uint8

Sub-Index	**00h**
Description	Number of assigned PDOs
Access	ro
PDO Mapping	no.
Value Range	0 (no PDO assigned to this channel)
Default Value	0 (no PDO assigned to this channel)

The name "Number of assigned PDOs" assigned by the EtherCAT specification for Subindex 0 of these objects is confusing here, as Sync Manager channels 0 and 1 are always allocated to the mailbox telegram. SDOs are always transmitted in this telegram type under EtherCAT CoE. Subindex 0 of these two objects is therefore unused.

Object 1C11h - Sync Manager Channel 1 (Mailbox Send)

This object allows a PDO to be configured for Sync channel 1. As Sync channel 1 is always allocated to the mailbox telegram protocol, the user cannot change this object. The object therefore always has the following values:

Index	**1C11h**
Name	Sync Manager Channel 1 (Mailbox Send)
Object Code	Array
Data Type	uint8

Sub-Index	**00h**
Description	Number of assigned PDOs
Access	ro
PDO Mapping	no.
Value Range	0 (no PDO assigned to this channel)
Default Value	0 (no PDO assigned to this channel)

Object 1C12h - Sync Manager Channel 2 (Process Data Output)

This object allows a PDO to be configured for Sync channel 2. Sync channel 2 is permanently assigned for the reception of Receive PDOs (Master => Slave). In this object, the number of PDOs allocated to this Sync channel must be set under sub-index 0.

The object number of the PDO to be allocated to the channel is subsequently entered in subindices 1 to 4. Only the object numbers of the previously configured Receive PDOs can be used for this (object 1600h to 1603h).

In the current implementation, the data of the objects below is not evaluated further by the firmware of the motor controller.

The CANopen configuration of the PDOs is used for evaluation under EtherCAT.

Index	**1C12h**
Name	Sync Manager Channel 2 (Process Data Output)
Object Code	Array
Data Type	uint8

Sub-Index	**00h**
Description	Number of assigned PDOs
Access	rw
PDO Mapping	no.
Value Range	0: no PDO assigned to this channel 1: one PDO assigned to this channel 2: two PDOs assigned to this channel 3: three PDOs assigned to this channel 4: four PDOs assigned to this channel
Default Value	0 :no PDO assigned to this channel

Sub-Index	**01h**
Description	PDO Mapping object Number of assigned RxPDO
Access	rw
PDO Mapping	no.
Value Range	0x1600: first Receive PDO
Default Value	0x1600: first Receive PDO

Sub-Index	**02h**
Description	PDO Mapping object Number of assigned RxPDO
Access	rw
PDO Mapping	no.
Value Range	0x1601: first Receive PDO
Default Value	0x1601: first Receive PDO

Sub-Index	**03h**
Description	PDO Mapping object Number of assigned RxPDO
Access	rw
PDO Mapping	no.
Value Range	0x1602: first Receive PDO
Default Value	0x1602: first Receive PDO

Sub-Index	**04h**
Description	PDO Mapping object Number of assigned RxPDO
Access	rw
PDO Mapping	no.
Value Range	0x1603: first Receive PDO
Default Value	0x1603: first Receive PDO

Object 1C13h - Sync Manager Channel 3 (Process Data Input)

This object allows a PDO to be configured for Sync channel 3. Sync channel 3 is permanently assigned for sending Transmit PDOs (Master <= Slave). In this object, the number of PDOs assigned to this Sync channel must be set under subindex 0.

The object number of the PDO to be allocated to the channel is subsequently entered in subindices 1 to 4. Only the object numbers of the previously configured Transmit PDOs can be used for this (0x1A00 to 0x1A03).

In the current implementation, the data of the objects below is not evaluated further by the firmware of the motor controller. The CANopen configuration of the PDOs is used for evaluation under EtherCAT.

Index	**1C13h**
Name	Sync Manager Channel 3 (Process Data Input)
Object Code	Array
Data Type	uint8

Sub-Index	**00h**
Description	Number of assigned PDOs
Access	rw
PDO Mapping	no.
Value Range	0: no PDO assigned to this channel 1: one PDO assigned to this channel 2: two PDOs assigned to this channel 3: three PDOs assigned to this channel 4: four PDOs assigned to this channel
Default Value	0 :no PDO assigned to this channel

Sub-Index	**01h**
Description	PDO Mapping object Number of assigned TxPDO
Access	rw
PDO Mapping	no.
Value Range	0x1A00: first Transmit PDO
Default Value	0x1A00: first Transmit PDO

Sub-Index	**02h**
Description	PDO Mapping object Number of assigned TxPDO
Access	rw
PDO Mapping	no.
Value Range	0x1A01: first Transmit PDO
Default Value	0x1A01: first Transmit PDO

Sub-Index	**03h**
Description	PDO Mapping object Number of assigned TxPDO
Access	rw
PDO Mapping	no.
Value Range	0x1A02: first Transmit PDO
Default Value	0x1A02: first Transmit PDO

Sub-Index	04h
Description	PDO Mapping object Number of assigned TxPDO
Access	rw
PDO Mapping	no.
Value Range	0x1A03: first Transmit PDO
Default Value	0x1A03: first Transmit PDO

5.1.3 Unsupported CANopen communication objects under CoE

When connecting the CMMP-AS under “CANopen over EtherCAT”, some CANopen objects, which are available under a direct connection of the CMMP-AS via CANopen, are not supported. These objects are listed in the table below:

Identifier	Name	Meaning
0x1008	Manufacturer device name (string)	Device name (object is not available)
0x1009	Manufacturer hardware version (string)	HW version (object is not available)
0x100A	Manufacturer software version (string)	SW version (object is not available)
0x6089	position_notation_index	Specifies the number of decimal places for displaying the position values in the controller. The object is only available as a data container. The firmware is not evaluated further.
0x608A	position_dimension_index	Specifies the unit for displaying the position values in the controller. The object is only available as a data container. The firmware is not evaluated further.
0x608B	velocity_notation_index	Specifies the number of decimal places for displaying the velocity values in the controller. The object is only available as a data container. The firmware is not evaluated further.

Identifier	Name	Meaning
0x608C	velocity_dimension_index	Specifies the unit for displaying the velocity values in the controller. The object is only available as a data container. The firmware is not evaluated further.
0x608D	acceleration_notation_index	Specifies the number of decimal places for displaying the acceleration values in the controller. The object is only available as a data container. The firmware is not evaluated further.
0x608E	acceleration_dimension_index	Specifies the unit for displaying the acceleration values in the controller. The object is only available as a data container. The firmware is not evaluated further.

5.2 FHPP profile

The FHPP profile can be used as an alternative to the DS301/402 profile. All FHPP objects listed in the FHPP manual for type CMM... motor controllers can be used.

Brief introduction to FHPP

Festo has developed an optimised data profile especially tailored to the target applications for handling and positioning tasks, the “Festo Handling and Positioning Profile (FHPP).”
The FHPP enables uniform control and programming for the various fieldbus systems and controllers from Festo.

In addition, it defines the following largely uniform settings for the user

– the operating modes,

– I/O data structure,

– parameter objects,

– sequence control.

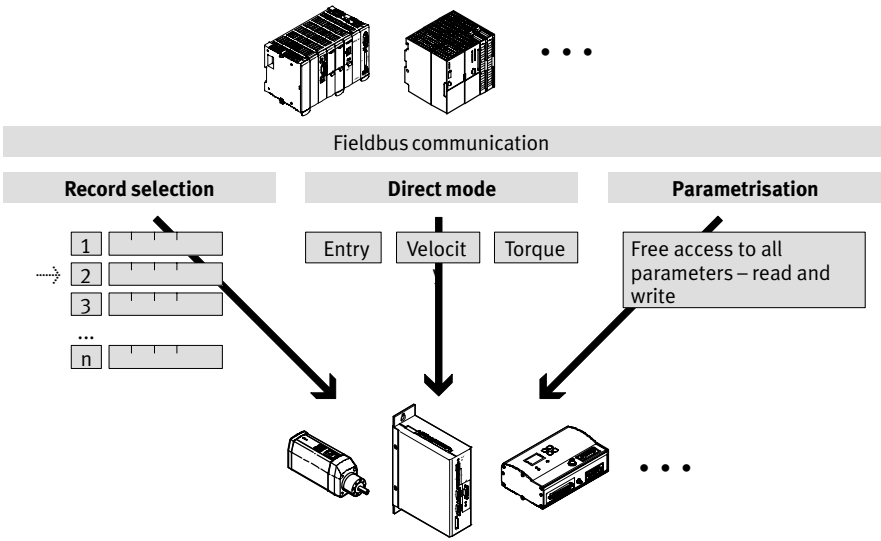

Fig. 5/6: The FHPP principle

Control and status data

Communication over the fieldbus is effected by way of 8-byte control and status data. Functions and status messages required in operation can be written and read directly.

Record selection

Saved positioning records can be processed in the Record Select mode.
For this purpose, positioning records are parametrised with the Festo Configuration Tool or taught via FHPP during commissioning.

Direct mode

In the Direct operating mode, the important positioning data is transferred directly via the control bytes.

– Target positions, velocities and torques can be ascertained and specified by the controller during run-time, depending on the operating status.

– There are no limitations relating to the number of saved positioning records.

Parametrisation

The controller can access all parameter values of the controller via the fieldbus by means of the parameter channel. A further 8 bytes of I/O data are used for this purpose.

Concept

A PLC exchanges the following data with the FHPP:

- 8-byte control and status data:
  - Control and status bytes,
  - Record number or setpoint position in the output data,
  - Feedback of actual position and record number in the input data,
  - Additional mode-dependent setpoint and actual values,
- If required, an additional 8 bytes of input and 8 bytes of output data for FPC parametrisation.

The FHPP protocol stipulates 8 bytes of input data and 8 bytes of output data. Of these, the first byte is fixed (the first two bytes in the FHPP operation modes Record Select and Direct mode). It remains intact in each operation mode and controls the enabling of the controller and the FHPP operation modes. The other bytes are dependent on the FHPP operation mode that was selected. Additional control or status bytes and target and actual values can be transmitted here.

In the cyclic data, a further 8 bytes of input data and 8 bytes of output data are permissible to transmit parameters according to the FPC protocol.

If applicable, observe the specification in the bus master for the representation of words and double words (Intel/ Motorola). E.g. in the ”little endian” representation when transmitted via CAN (lower-value byte first).

FHPP with EtherCAT

The FHPP data is distributed to two process data objects for EtherCAT communication. The CoE protocol is also used for transmission.

Supported process data objects	**Fixed data mapping of the FHPP data**
TxPDO 1	FHPP standard 8 byte control data
TxPDO 2	FPC parameter channel Request to read/write FHPP parameter values
RxPDO 1	FHPP standard 8 byte status data
RxPDO 2	FPC parameter channel Transmission of requested FHPP parameter values

Refer to the following manual for a detailed description of the FHPP data profile:
“Festo Handling and Positioning Profile”
P.BE-CMM-FHPP-SW-....

5.3 Communication finite state machine

As in almost all fieldbus interfaces for motor controllers, the connected slave (in this case the CMMP-AS motor controller) must first be initialised by the master before it can be used by the master in an application. For this purpose, a finite state machine is required for communication, to specify a fixed sequence of actions for this initialisation process.

A finite state machine is also defined for the EtherCAT interface. Changes between the individual statuses of the finite state machine may only occur between specific statuses, and are always initiated by the master. Slaves may not implement status changes independently. The individual statuses and the permitted status changes are described in the following tables and figures.

Status	Description
Power ON	The device was switched on. It initialises itself and switches directly to the “Init” status.
Init	In this status, the EtherCAT fieldbus is synchronised by the master. This includes setting up the asynchronous communication between master and slave (mailbox telegram protocol). There is no direct communication between the master and slave yet. The configuration starts, saved values are loaded. When all devices are connected to the bus and configured, the status switches to “Pre-Operational”.
Pre-Operational	In this status, asynchronous communication between the master and slave is active. The master uses this status to set up possible cyclic communication via PDOs and use acyclic communication for parametrisation. If this status runs without errors, the master switches to the “Safe-Operational” status.

Status	Description
Safe-Operational	This status is used to set the devices connected to the EtherCAT bus to a safe status. The slave sends up-to-date actual values to the master, but ignores new setpoint values from the master and uses safe default values instead. If this status runs without errors, the master switches to the “Operational” status.
Operational	In this status, both acyclic and cyclic communication are active. Masters and slaves exchange target and actual value data. In this status, the CMMP-AS can be enabled and travel via the CoE protocol.

Only transitions in accordance with Fig. 5/7 are permitted between the individual status of the communication finite state machine:

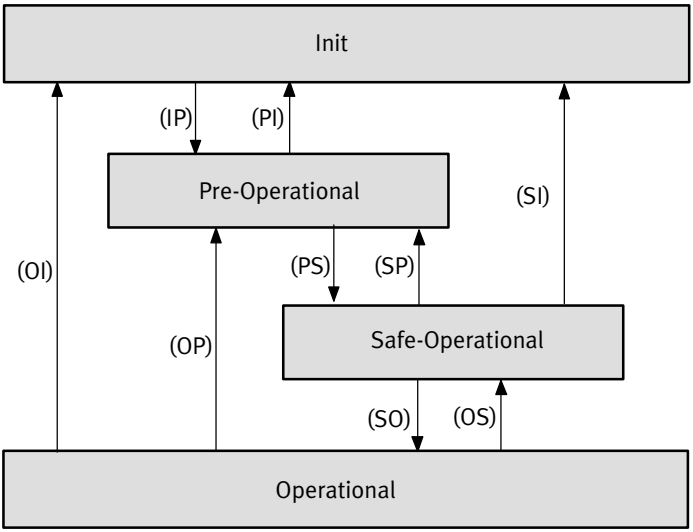

Fig. 5/7: Communication finite state machine

The transitions are described individually in the following table.

Status transition	Status
IP	Start of acyclic communication (mailbox telegram protocol)
PI	Stop of acyclic communication (mailbox telegram protocol)
PS	Start Inputs Update: Start of cyclic communication (process data telegram protocol) Slave sends actual values to master. The slave ignores setpoint values from the master and uses internal default values.
SP	Stop Input Update: Stop of cyclic communication (process data telegram protocol). The slave no longer sends actual values to the master.
SO	Start Output Update: The slave evaluates up-to-date setpoint specifications from the master.
OS	Stop Output Update: The slave ignores setpoint values from the master and uses internal default values.
OP	Stop Output Update, Stop Input Update: Stop of cyclic communication (process data telegram protocol). The slave no longer sends actual values to the master and the master no longer sends setpoint values to the slave.
SI	Stop Input Update, Stop Mailbox Communication: Stop of cyclic communication (process data telegram protocol) and stop of acyclic communication (mailbox telegram protocol). The slave no longer sends actual values to the master and the master no longer sends setpoint values to the slave.
OI	Stop Output Update , Stop Input Update, Stop Mailbox Communication: Stop of cyclic communication (process data telegram protocol) and stop of acyclic communication (mailbox telegram protocol). The slave no longer sends actual values to the master and the master no longer sends setpoint values to the slave.

In the EtherCAT finite state machine, the “Bootstrap” status is also specified in addition to the statuses listed here. This status is intended for loading a new firmware to the slave when the EtherCAT protocol is running.
As the firmware is downloaded via the RS232 interface, this status is not implemented for the CMMP-AS.

5.3.1 Differences between the finite state machines of CANopen and EtherCAT

When operating the CMMP-AS via the EtherCAT-CoE protocol, the EtherCAT finite state machine is used instead of the CANopen NMT finite state machine. This differs from the CANopen finite state machine in several aspects. These different characteristics are listed below:

– No direct transition from Pre-Operational after Power On

– No Stopped status, direct transition to the INIT status

– Additional status: Safe-Operational

The following table compares the different statuses:

EtherCAT State	**CANopen NMT State**
Power ON	Power-On (initialisation)
Init	Stopped
Safe-Operational	---
Operational	Operational

5.4 SDO Frame

All data of an SDO transfer are transmitted via SDO frames in CoE. These frames have the following structure:

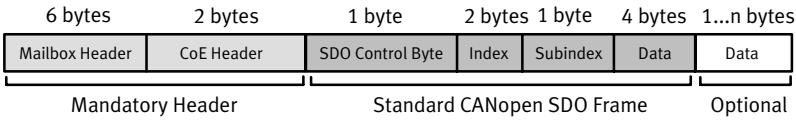

Fig. 5/8: SDO Frame: Telegram structure

Mailbox Header	Data for mail box communication (length, address and type)
CoE Header	Identifier of the CoE service
SDO Control Byte	Identifier for a read or write command
Index	Main index of the CANopen communication object
Subindex	Subindex of the CANopen communication object
Subindex	Subindex of the CANopen communication object
Data (optional)	Additional optional data. This option is not supported by the CMMP-AS motor controller, as only standard CANopen objects can be addressed. The maximum size of these objects is 32 bits.

In order to transmit a standard CANopen object via one of these SDO frames, the actual CANopen SDO frame is packaged in an EtherCAT SDO frame and transmitted.

Standard CANopen SDO frames can be used for:

- Initialisation of the SDO download
- Download of the SDO segment
- Initialisation of the SDO upload
- Upload of the SDO segment
- Abort of the SDO transfer
- SDO upload expedited request
- SDO upload expedited response
- SDO upload segmented request (max. 1 segment with 4 bytes of user data)
- SDO upload segmented response (max. 1 segment with 4 bytes of user data)

All above-mentioned transfer types are supported by the CMMP-AS motor controller.
As the use of the CoE implementation of the CMMP-AS only allows the standard CANopen objects to be addressed, whose size is restricted to 32 bits (4 bytes), only transfer types with a maximum data length of up to 32 bits (4 bytes) are supported.

5.5 PDO Frame

Process Data Objects (PDO) are used for cyclic transmission of setpoint values and actual values between the masters and slaves. They must be configured in the “Pre-Operational” status by the master before the slave is operated. They are then transmitted in PDO frames. These PDO frames have the following structure:

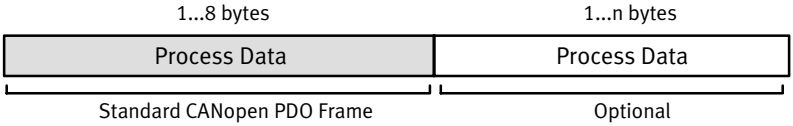

Fig. 5/9: PDO Frame: Telegram structure

Process Data	Data content of the PDO (Process Data Object)
Process Data (optional)	Optional data content of additional PDOs

To transmit a PDO via the EtherCAT CoE protocol, the transmit and receive PDOs must also be allocated to a transfer channel of the Sync Manager in addition to the PDO configuration (PDO Mapping) (see Chapter 6.1.1 “Configuring the communication interface”). The data of PDOs for the CMMP-AS motor controller is exchanged exclusively via the EtherCAT process data telegram protocol.

The transfer of CANopen process data (PDOs) via acyclic communication (mailbox telegram protocol) is not supported by the CMMP-AS motor controller.

As all data exchanged via the EtherCAT CoE protocol is forwarded internally to the CANopen implementation in the CMMP-AS motor controller, the PDO mapping is also implemented as described in the “CMMP-AS motor controller” CANopen manual. The figure below depicts this process:

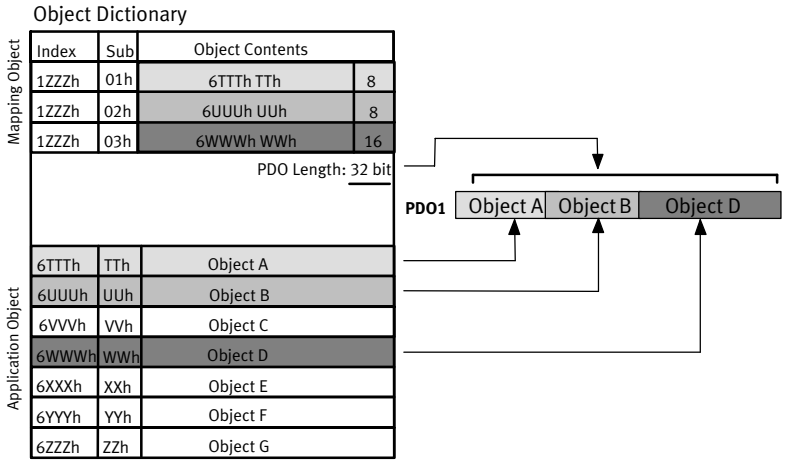

Fig. 5/10: PDO mapping

The simple forwarding of the data received via CoE to the CANopen protocol implemented in CMMP-AS means that the “Transmission Types” of the PDOs available for the CANopen protocol for the CMMP-AS can be used in addition to CANopen object mapping for the PDOs to be parametrised.

The CMMP-AS motor controller also supports the “Sync Message” transmission type. However, the Sync Message does not have to be sent via EtherCAT.

It is used either for the arrival of the telegram or the hardware synchronisation pulse of the “Distributed Clocks” mechanism (see below) for data transfer.
The EtherCAT interface for CMMP-AS supports synchronisation via the “Distributed Clocks” mechanism specified under EtherCAT by means of the use of FPGA module ESC20. The current regulator of the CMMP-AS motor controller is synchronised to this pulse, and the PDOs configured accordingly are evaluated or sent.

Refer to the CANopen manual of the CMMP-AS motor controller for more details.

The CMMP-AS motor controller with the EtherCAT interface supports the following functions:

– Cyclic PDO frame telegram via the process data telegram protocol.

– Synchronous PDO frame telegram via the process data telegram protocol.

The CMMP-AS motor controller with the EtherCAT interface supports four Receive PDOs (RxPDO) and four Transmit PDOs (TxPDO).

5.6 Error Control

The EtherCAT CoE implementation for the CMMP-AS motor controller monitors the following error statuses of the EtherCAT fieldbus:

- FPGA is not ready when the system is started.
- A bus error has occurred.
- An error has occurred on the mailbox channel. The following errors are monitored in this case:
  - An unknown service is requested.
  - A protocol other than CANopen over EtherCAT (CoE) is to be used.
  - An unknown Sync Manager is addressed.

All of these errors are defined as corresponding error codes for the CMMP-AS motor controller. If one of the abovementioned errors occurs, it is transmitted to the controller via a “Standard Emergency Frame”. See also the 5.7 “Emergency Frame” chapter and 7.1 ”Service functions and error messages” chapter.

The CMMP-AS motor controller with the EtherCAT interface supports the following function:

- Application Controller determines a defined error message number as a result of an event (Error Control Frame Telegram from the controller).

5.7 Emergency Frame

The master and slaves exchange error messages via the EtherCAT CoE emergency frame. The CoE emergency frames are used for direct transfer of the “Emergency Messages” defined under CANopen. The CANopen telegrams are simply tunnelled through the CoE emergency frames, as is the case for SDO and PDO transmission.

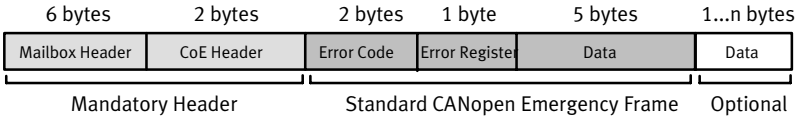

Fig. 5/11: Emergency Frame: Telegram structure

Mailbox Header	Data for mailbox communication (length, address and type)
CoE Header	Identifier of the CoE service
ErrorCode	Error Code of the CANopen EMERGENCY Message
ErrorCode	Error Code of the CANopen EMERGENCY Message
Data	Data content of the CANopen EMERGENCY Message
Data (optional)	Additional optional data. As only the standard CANopen emergency frames are supported in the CoE implementation for the CMMP-AS motor controller, the “Data (optional)” field is not supported.

As the “Emergency Messages” received and sent via CoE are simply forwarded to the CANopen protocol implemented in the motor controller, all error messages can be looked up in the CANopen manual for the CMMP motor controller.

5.8 Adaptation of the XML device description file

As already mentioned in the 2.5 “XML description file” section, every device is described in a device description file under EtherCAT. This file can be used for simple connection of the EtherCAT devices to an EtherCAT controller. This file contains the complete parametrisation of the slave, including the parametrisation of the Sync Manager and the PDOs. For this reason, the configuration of the slave can be changed using this file.

Festo has created a device description file for the CMMP-AS motor controller. It can be downloaded from the Festo homepage. Its contents will now be described in more detail to permit users to adapt the file to their application.

The available device description file supports both the DS402 profile and the FHPP via separately selectable modules.

5.8.1 Fundamental structure of the device description file

The EtherCAT device description file is in the XML format. This format has the advantage that it can be read and edited in a standard text editor. XML files always describe a tree structure. It defines the individual branches via nodes. These nodes have start and end flags. Each node can contain any number of sub-nodes.

EXAMPLE: Rough explanation of the fundamental structure of an XML file:

```
‹car›
    ‹engine›
        ‹pistons›4‹/pistons›
        ‹valves›8‹/valves›
        ‹power›78 kW‹/power›
        ...
    ‹/engine›
    ‹chassis›
        ‹interior›
        ...
        ‹/interior›
        ...
    ‹/chassis›
    ...
‹/car›
```

The following brief rules must be observed for the structure of an XML file:

– Each node must have a unique name.

– Each node opens with ‹node name› and closes with ‹/node name›.

The device description file for the CMMP-AS motor controller under EtherCAT CoE is divided into the following sub-items:

Node name	Meaning	Adaptable
Vendor	This node contains the name and the ID of the manufacturer of the device to which this description file belongs. It also contains the binary code of a bitmap with the manufacturer's logo.	No
Description	This sub-item contains the actual device description including the configuration and initialisation.	Partial
Group	This node contains the assignment of the device to a device group. These groups are specified and may not be changed by the user.	No
Devices	This sub-item contains the actual description of the device.	Partial

The following table describes only the subnodes of the "Description" node, required for the parametrisation of the CMMP-AS motor controller under CoE. All other nodes are fixed and may not be changed by the user.

Node name	Meaning	Adaptable
RxPDO Fixed=...	This node contains the PDO Mapping and the assignment of the PDO to the Sync Manager for Receive PDOs.	Yes
TxPDO Fixed=...	This node contains the PDO Mapping and the assignment of the PDO to the Sync Manager for Transmit PDOs.	Yes
Mailbox	This node allows commands, which are transmitted to the slave via SDO transfers by the master during the phase transition from "Pre-Operational" to "Operational", to be defined	Yes

As only the nodes from the table above are important for users to adapt the device description file, they are described in detail in the following chapters. The remaining content of the device description file is fixed and may not be changed by the user.

Caution
If changes are made to nodes and content other than the RxPDO, TxPDO and Mailbox nodes in the device description file, error-free operation of the device can no longer be guaranteed.

5.8.2 Receive PDO configuration in the RxPDO node

The RxPDO node is used to specify the mapping for the Receive PDOs and their assignment to a channel of the Sync Manager. A typical entry in the device description file for the CMMP-AS motor controller can be as follows:

```
‹RxPDO Fixed=”1” Sm=”2”›
    ‹Index›#x1600‹/Index›
    ‹Name›Outputs‹/Name›
    ‹Entry›
        ‹Index›#x6040‹/Index›
        ‹SubIndex›0‹/SubIndex›
        ‹BitLen›16‹/BitLen›
        ‹Name›Controlword‹/Name›
        ‹DataType›UINT‹/DataType›
    ‹/Entry›
    ‹Entry›
        ‹Index›#x6060‹/Index›
        ‹SubIndex›0‹/SubIndex›
        ‹BitLen›8‹/BitLen›
        ‹Name›Mode_Of_Operation‹/Name›
        ‹DataType›USINT‹/DataType›
    ‹/Entry›
‹/RxPDO›
```

As the example above shows, the entire mapping of the Receive PDO is described in detail in such entries. The first large block specifies the object number of the PDO and its type. This is followed by a list of all CANopen objects which are to be mapped to the PDO.

The table below describes the individual entries in greater detail:

Node name	Meaning	Adaptable
RxPDO Fixed="1" Sm="2"	This node describes the properties of the Receive PDO directly and its assignment to the Sync Manager. The Fixed="1" entry indicates that the object mapping cannot be changed. The Sm="2" entry indicates that the PDO is to be allocated to Sync channel 2 of the Sync Manager.	No
Index	This entry contains the object number of the PDO. The first Receive PDO under object number 0x1600 is configured here.	Yes
Name	The name indicates that this PDO is a Receive PDO (outputs) or a Transmit PDO (inputs). This value must always be set to "Output" for a Receive PDO.	No
Entry	Each Entry node contains a CANopen object to be mapped to the PDO. An Entry node contains the index and subindex of the CANopen object to be mapped, as well as their name and data type.	Yes

The sequence and mapping of the individual CANopen objects for the PDO correspond to the sequence in which they are specified via the “Entry” entries in the device description file. The individual sub-items of an “Entry” node are specified in the following table:

Node name	Meaning	Adaptable
Index	This entry specifies the index of the CANopen object to be mapped to the PDO.	Yes
Subindex	This entry specifies the subindex of the CANopen object to be mapped.	Yes
BitLen	This entry specifies the size of the object to be mapped in bits. This entry must always correspond with the type of the object to be mapped. Permitted: 8 bits / 16 bits / 32 bits.	Yes
Name	This entry specifies the name of the object to be mapped as a string.	Yes
Data Type	This entry specifies the data type of the object to be mapped. See the CANopen manual for the CMMP motor controller for the data types of the individual CANopen objects.	Yes

5.8.3 Transmit PDO configuration in the TxPDO node

The TxPDO node is used to specify the mapping for the Transmit PDOs and their assignment to a channel of the Sync Manager. The configuration is the same as that for Receive PDOs as described in section 5.8.2 “Receive PDO configuration in the RxPDO node”, except that the “Name” node of the PDO must be set to “Inputs” instead of ”Outputs”.

5.8.4 Initialisation commands via the “Mailbox” node

The “Mailbox” node in the device description file is used to describe CANopen objects via the master in the slave during the initialisation phase. The commands and objects to be described there are specified via special entries. These entries specify the phase transition at which this value is to be written. Such entries contain the CANopen object number (index and subindex) as well as the data value to be written and a comment.

A typical entry has the following form:

```
‹InitCmd›
    ‹Transition›PS‹/Transition›
    ‹Index#x6060‹/Index›
    ‹SubIndex›0‹/SubIndex›
    ‹Data›03‹/Data›
    ‹Comment›velocity mode‹/Comment›
‹/InitCmd›
```

In the example above, status transition PS from “Pre-Operational” to “Safe Operational” sets the operating mode in the CANopen object “modes_of_operation” to “Speed adjustment”. The individual subnodes have the following meaning:

Node name	**Meaning**	**Adaptable**
Transition	Name of the status transition at which the command is to be run (see Chapter 6.2 “Communication finite state machine” on this).	Yes
Index	Index of the CANopen object to be written	Yes
Subindex	Subindex of the CANopen object to be written	Yes
Data	Data value to be written, as a hexadecimal value	Yes
Comment	Comment on this command	Yes

Caution
In a device description file for the CMMP-AS motor controller, some entries in this section are pre-assigned. These entries must remain as they are and may not be changed by the user.

5.9 Synchronisation (Distributed Clocks)

Time synchronisation is implemented via so-called “Distributed Clocks” in EtherCAT. Each EtherCAT slave receives a real-time clock, which is synchronise in all slaves by the master during the initialisation phase. The clocks in all slaves are then adjusted during operation.

This provides a uniform time base in the entire system with which the individual slaves can synchronise. The Sync telegrams provided for this purpose under CANopen are unnecessary under CoE.

The FPGA ESC20 used in the CMMP-AS motor controller supports Distributed Clocks. This facilitates extremely precise time synchronisation. The cycle time of the EtherCAT frame must match the cycle time tp of the controller-internal interpolator exactly. If necessary, the interpolator time must be adjusted via the object in the device description file.

In the present implementation, synchronous transfer of PDO data and synchronisation of the controller-internal PLL to the synchronous data framework of the EtherCAT frame can be implemented even without Distributed Clocks. For this purpose, the firmware uses the arrival of the EtherCAT frame as a time base.

The following restrictions apply:

– The master must be able to send the EtherCAT frames with an extremely low jitter.

– The cycle time of the EtherCAT frame must match the cycle time tp of the controller-internal interpolator exactly.

– The Ethernet must be available exclusively for the EtherCAT frame. It may be necessary to synchronise other telegrams to the grid, as they may not block the bus.

Commissioning

Chapter 6

Contents

6. Commissioning 6-1

6.1 Introduction 6-3
6.2 Requirements 6-4
6.3 Important note 6-5
6.4 Step by step 6-6
6.4.1 Configuration using FCT 6-6
6.5 Configuration on a Beckhoff PLC 6-9
6.5.1 Configuration of process data 6-13

6.1 Introduction

This section is intended to provide examples for CMMP-AS configuration for operation on a Beckhoff PLC in an EtherCAT network.

The following components were used for sample parametrisation:

- CMMP-AS-C2-3A: Servo controller
- CAMC-EC: EtherCAT module for the CMMP-AS
- EMMS-AS-55-S-TS: Servo motor
- Beckhoff CX1020-0113: Beckhoff PLC
- Beckhoff CX1100-0004: Beckhoff power supply unit for PLC
- Beckhoff EK1110: Beckhoff EtherCAT extension module
- Ethernet CAT5e patch cable

6.2 Requirements

- PC with Windows 2000/XP/Vista
  - Installed FCT framework (Version 1.1.5.120)
  - Installed FCT plug-in for CMMP-AS (Version 1.2.0.5xx)
  - Installed Beckhoff TwinCAT (tested with TwinCAT v2.10.0 Build 1340)
  - XML description file for the CMMP-AS
- CMMP-AS-...* Servo controller
- EMMS-AS-... * Servo motor
  with NEBM-...* Motor and encoder cable
- CAMC-EC EtherCAT module
- 24V DC power supply unit
- Cable for the power supply *

*) Components depend on the application

6.3 Important note

Ensure that the target system (PLC) contains at least Version 2.10 (Build 1328)! Otherwise, the automatic download of the PDO configuration will not function.

To find the most up-to-date version for your PLC, please contact the Beckhoff technical support team or search this internet site:
ftp://ftp.beckhoff.com/Software/embPC-Control/

6.4 Step by step

Connect the servo motor (EMMS-AS-...) via the motor/encoder cable (NEBM-...) to the servo controller (CMMP-AS-) as described in the corresponding manuals. Insert the EtherCAT module (CAMC-EC) in the Ext-2 slot of the CMMP-AS as shown in the corresponding description.

6.4.1 Configuration using FCT

This section only describes the steps which directly concern the EtherCAT fieldbus. Start FCT and create a new project. If you also have older versions of CMMP-AS on your computer ensure that you select Version V 1.2.0 for the new project!

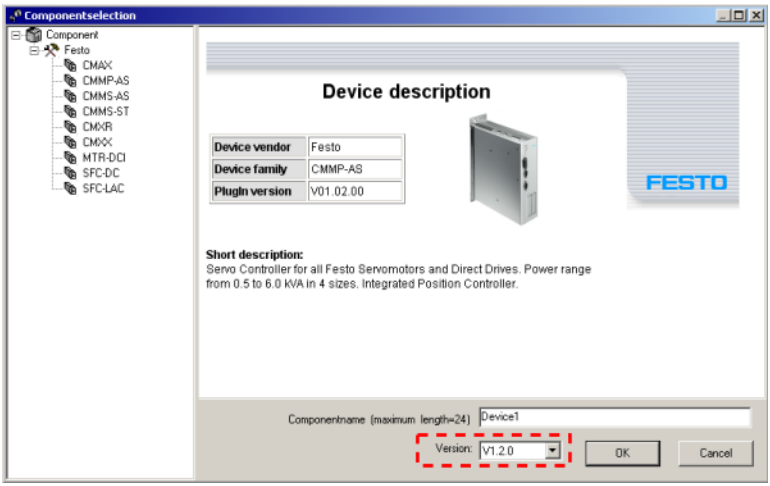

Fig. 6/12: FCT 1

Under Option Slot 2 on the “Configuration” page, select “CAMC-EC: EtherCAT”.

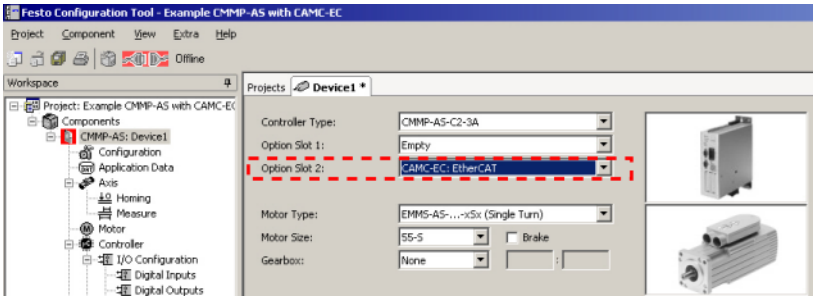

Fig. 6/13: FCT 2

The EtherCAT control interface is automatically selected on the “Application data” page and this selection cannot be changed.

You can set the required data profile (CANopen DS402 or Festo FHPP) on the “FieldBus” page. You can also activate the “Factor Group” in the second tab on the FieldBus page. This converts the position, speed and acceleration values to normal units. If this option is not activated, all values are specified or read in increments.

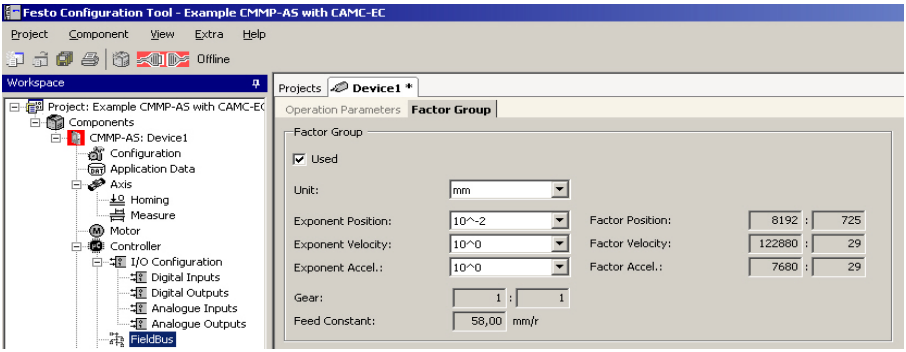

Fig. 6/14: FCT 3

When all parameters required for the application have been set, start a download. This transmits the data from FCT to CMMP-AS. Then run “Save”, to save the data just transferred permanently in the device. Shut down the power supply to the logic unit (24V) and then reactivate it to reboot the system.

6.5 Configuration on a Beckhoff PLC

This point only describes the hardware connection of the CMMP to the Beckhoff PLC. I.e. an application program with “TwinCAT PLC Control” must already have been created before all of these steps are performed.

Before you start, copy the XML device description file (Festo_CMMP-AS_Vx.xml) to the TwinCAT software installation folder. This is generally “C:\TwinCAT\Io\EtherCAT\”.

Start the TwinCAT System Manager and create a new configuration. Under SYSTEM - Configuration click the “Choose Target”, to select the PLC which is the EtherCAT master.

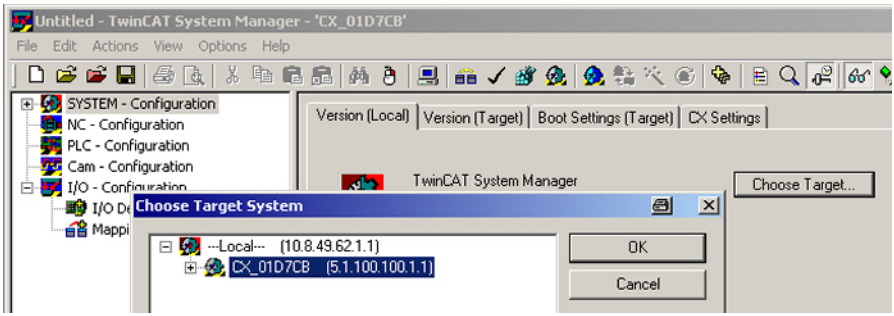

Fig. 6/15: TwinCAT 1

Right click the ”PLC - Configuration” item to read the application program.

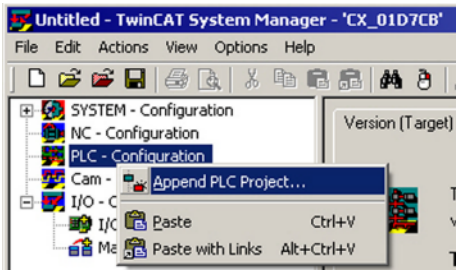

Fig. 6/16: TwinCAT 2

Now start the PLC in config mode.

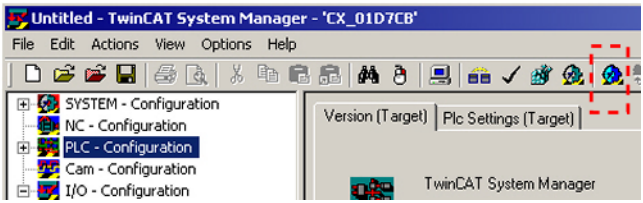

Fig. 6/17: Config mode 1

When the PLC has started in Config mode, right-click I/O Devices to search for all devices connected to the PLC (e.g. fieldbus masters).

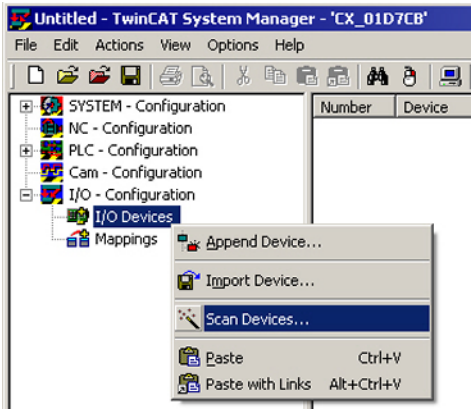

Fig. 6/18: Config mode 2

After the scan is completed, an overview of all found devices is displayed.

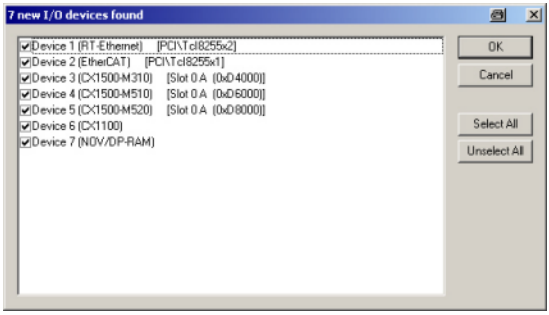

Fig. 6/19: Overview of found devices

The next dialog box prompts you to search for participants connected to the devices previously found. Click Yes to confirm, and the connected CMMP-AS unit(s) should appear under the “EtherCAT” device.

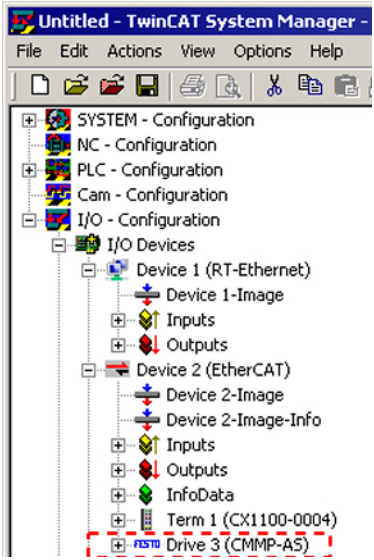

Fig. 6/20: CMMP-AS under EtherCAT

6.5.1 Configuration of process data

Festo FHPP

From the list, select the CMMP-AS which is to be operated via the FHPP profile, then go to the ”Process Data” tab.

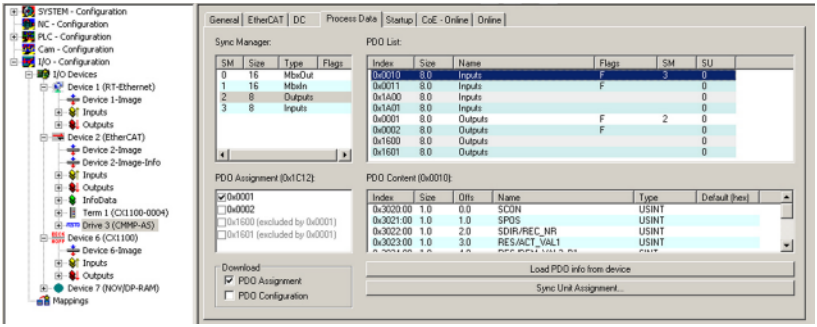

Fig. 6/21: Process data

In the default settings of the XML device description file, module 0x0001 is activated in the PDO assignment for the output channel and module 0x0010 is activated in the PDO assignment for the input channel of the Sync Manager. These modules (0x0001 and 0x0010) allow the CMMP-AS to be operated via FHPP.

If you also want to use FPC (Festo Parameter Channel) to allow further parameters to be adjusted for subsequent operation, you must also activate the 0x0002 module in the output channel and 0x0010 module in the input channel of the Sync Manager.

Leave the settings in the “Download” field as they are (PDO Assignment activated, PDO Configuration deactivated).

Link the activated variables to the variables from the previously integrated PLC programme.

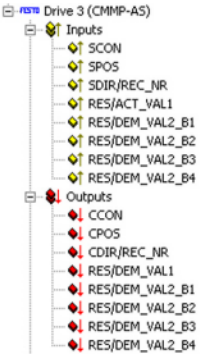

Fig. 6/22: Variable assignment

CANopen DS402

From the list, select the CMMP-AS which is to be operated via the CANopen DS402 profile, then go to the “Process Data” tab.

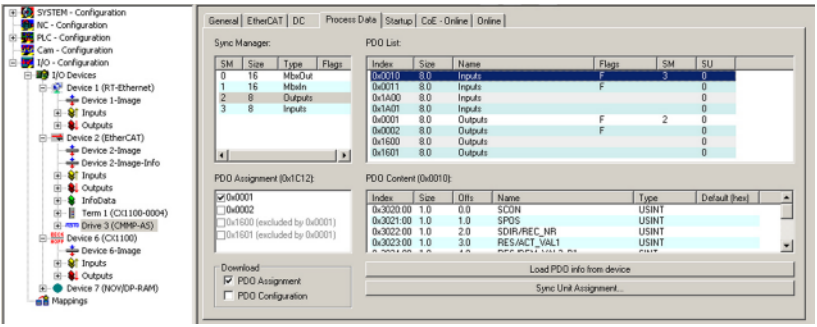

Fig. 6/23: Process data

As the modules for FHPP are automatically activated in the default settings of the XML device description file, you must now make changes in the PDO assignment to enable the modules for DS402 operation.

To do so, deactivate the 0x0001 module in the PDO assignment of the output channel and the 0x0010 module in the input channel of the Sync Manager.

The 0x1600 and 0x1601 modules for the output channel and 0x1A00 and 0x1A01 modules for the input channel can now be activated. This disables the modules for FHPP to prevent data inconsistency.

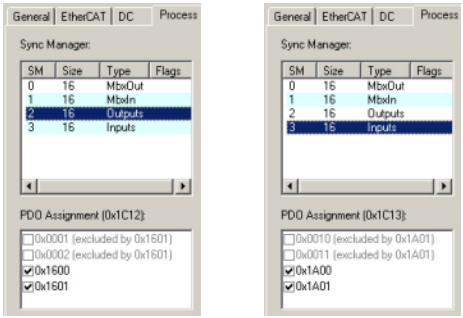

Fig. 6/24: PDO assignment 1

In addition, it is also essential that you change the settings in the "Download" field, to allow the PDO data to be downloaded automatically (PDO Assignment activated, PDO Configuration activated).

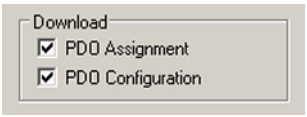

Fig. 6/25: PDO assignment 2

Link the activated variables to the variables from the previously integrated PLC programme.

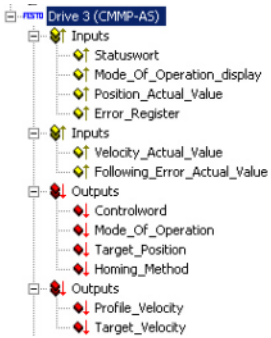

Fig. 6/26: Variable assignment

Service functions and error messages

Chapter 7

Contents

7. Service functions and error messages 7-1

7.1 Operating mode and error messages 7-3
7.1.1 Error messages 7-3

7.1 Operating mode and error messages

7.1.1 Error messages

When an error occurs, the CMMP-AS motor controller displays an error message in the parametrisation software.

See the user manual for further details on other error messages.

The table below summarises the specific EtherCAT error messages:

Error message	Meaning	Measures
62-0	EtherCAT - General bus error	Switch on the EtherCAT master. Check the cabling.
62-1	EtherCAT - ESC chip not detected	Hardware fault: Contact the technical support team.
62-2	EtherCAT protocol error, not CoE	Incorrect protocol. CAN bus cabling fault.
62-3	EtherCAT protocol error, RPDO SM2 length not ok	Check the RPDO configuration of the CMMP and the controller
62-4	EtherCAT protocol error, TPDO SM3 length not ok	Check the TPDO configuration of the CMMP and the controller.
62-5	EtherCAT - Synchronous transmission not ok	Check the configuration of the master. Synchronous transfer is unstable.
63-0	No EtherCAT-ESC20 chip available	Hardware fault: Please contact the technical support team.
63-1	Faulty data on the EtherCAT bus	Check cabling.

Error message	Meaning	Measures
63-2	TPDO data not read.	The data was sent faster than the motor controller could process it. Reduce the cycle time on the EtherCAT bus.
63-3	When the EtherCAT was started, no hardware SYNC (Distributed Clocks) was found. The firmware now synchronises on the EtherCAT frame.	If necessary, check whether the master supports the “Distributed Clocks” property. If not: Ensure that the EtherCAT frames are not interrupted by other frames if the “interpolated position mode” is to be used.

Tab. 7/7: Error messages

Technical Appendix

Appendix A

Contents

A. Technical Appendix A-1

A.1 Technical data A-3

A.1 Technical data

Area	Values
Storage temperature range	-25°C to +75°C
Operating temperature range	0°C to 50°C
Humidity	0 ... 90%, non-condensing
Installation height	Up to 1000 m above sea level
External dimensions (LxWxH):	Approx. 92 x 65 x 19 mm
Weight	Approx. 55 g
Slot	Option slot Ext2 only
EtherCAT interface, signal level	0 ... 2.5 VDC
EtherCAT interface, differential voltage	1,9 ... 2.1 VDC